

CLAIMS

What is claimed is:

1. A flexible operating system comprising:
operability for executing in a first manner as a native operating system on a computer system and for executing in a second manner as a virtualized operating system on said computer system; and
code for determining whether said operating system is being used as a native operating system or as a virtualized operating system on said computer system.
2. The flexible operating system of claim 1 further comprising:
code for selectively executing in said first manner if determined that said operating system is being used as a native operating system by said computer system and in said second manner if determined that said operating system is being used as a virtualized operating system on said computer system.
3. The flexible operating system of claim 2 wherein said second manner comprises acting as a paravirtualized operating system.
4. The flexible operating system of claim 3 wherein said paravirtualized operating system is operable to make a call to a Virtual Machine Monitor (VMM) for performing at least one privileged operation.
5. The flexible operating system of claim 1 wherein said code for determining comprises:
code for checking a global variable that indicates whether said operating system is being used as a native operating system or as a virtualized operating system on said computer system.
6. The flexible operating system of claim 5 further comprising:
code for executing an instruction which, when the operating system is being used as a virtualized operating system causes a Virtual Machine Monitor (VMM) to set at least one configuration bit to a first value and when the operating system is being used as a native operating system causes the VMM to set said at least one configuration bit to a different value.

7. The flexible operating system of claim 6 further comprising:
code for setting said global variable based at least in part on the value of said at least one configuration bit after executing said instruction.

8. The flexible operating system of claim 1 further comprising:
code for making a call to a Virtual Machine Monitor (VMM) for performing at least one privileged operation.

9. The flexible operating system of claim 8 wherein said code for making a call to said VMM uses an Application Program Interface (API) defined for said VMM.

10. The flexible operating system of claim 8 wherein said code for making a call to said VMM is used for performing said at least one privileged operation if determined that said operating system is being used as virtualized operating system on said computer system.

11. A method comprising:
implementing at least one operating system on a computer system;
determining, by said computer system, whether said at least one operating system is a native operating system or a guest operating system on a virtual machine;
said at least one operating system operating in a first manner if determined that it is a native operating system; and
said at least one operating system operating in a second manner if determined that it is a guest operating system on a virtual machine.

12. The method of claim 11 wherein said determining comprises:
said at least one operating system determining whether it is being used as said native operating system or as said guest.

13. The method of claim 12 wherein said at least one operating system determines whether it is being used as said native operating system or as said guest based at least in part on a value of a global variable.

14. The method of claim 11 wherein said first manner comprises acting as a native operating system.

15. The method of claim 11 wherein said second manner comprises acting as a paravirtualized operating system.

16. The method of claim 15 wherein said paravirtualized operating system makes, for at least one privileged operation, a call to a Virtual Machine Monitor (VMM).

17. An operating system comprising:
code for determining whether said operating system is running virtualized; and
code for adapting operation of said operating system depending on whether it is running virtualized.

18. The operating system of claim 17 wherein said code for determining whether said operating system is running virtualized comprises code for checking the value of a global variable.

19. The operating system of claim 18 wherein said code for checking the value of a global variable checks said value of said global variable before performing certain privileged operations.

20. The operating system of claim 17 wherein said code for determining comprises:
code for determining, before execution of certain privileged instructions, whether said operating system is running virtualized.

21. The operating system of claim 20 wherein said code for adapting comprises:
if determined that said operating system is running virtualized, adapting operation of said operating system in executing said certain privileged instructions.

22. The operating system of claim 21 wherein said adapting operation of said operating system in executing said certain privileged instructions comprises:
making at least one call to a Virtual Machine Monitor (VMM).

23. The operating system of claim 17 wherein said code for adapting comprises:
code for a call to a Virtual Machine Monitor (VMM) for at least one privileged instruction.

24. The operating system of claim 17 wherein said code for determining comprises: code for executing an instruction which, when the operating system is being used as a virtualized operating system causes a Virtual Machine Monitor (VMM) to set at least one configuration bit to a first value.

25. The operating system of claim 24 wherein said code for determining further comprises:
code for determining whether said operating system is running virtualized based at least in part on a determined value of at least one configuration bit after execution of said instruction.

26. The operating system of claim 24 wherein said code for determining further comprises:
setting a global variable to a value based at least in part on the value of said at least one configuration bit after execution of said instruction.

27. A system comprising:
hardware resources;
at least one operating system for managing said hardware resources, wherein said at least one operating system is operable to determine whether it is running in a virtualized environment.

28. The system of claim 27 wherein said at least one operating system is operable to selectively execute in a first manner if determined that said at least one operating system is not running in a virtualized environment and in said second manner if determined that said at least one operating system is running in a virtualized environment.

29. The system of claim 28 wherein said first manner comprises acting as a native operating system.

30. The system of claim 28 wherein said second manner comprises acting as a paravirtualized operating system.

31. The system of claim 30 wherein said paravirtualized operating system is operable to make a call to a Virtual Machine Monitor (VMM) for performing at least one privileged operation.

32. The system of claim 27 further comprising:
a Virtual Machine Monitor (VMM).

33. The system of claim 32 wherein said at least one operating system adapts its operation to make a call to said VMM for performance of at least one privileged instruction when said at least one operating system determines that it is running in a virtualized environment.

34. The system of claim 27 wherein said virtualized environment comprises a Virtual Machine Monitor (VMM).

35. A system comprising:
a flexible operating system that is capable of acting as either a native operating system or as a virtualized operating system; and
means for determining whether the operating system is to be used as a native operating system or as a virtualized operating system, wherein the determining means stores information that is accessible by the flexible operating system to indicate whether the flexible operating system is being used as a native or as a virtualized operating system.

36. The system of claim 35 wherein the determining means makes the determination during a boot-up process of the system.

37. The system of claim 35 further comprising:
means for virtualizing resources of said system and multiplexing said resources among one or more virtualized operating systems.

38. The system of claim 35 wherein said flexible operating system is operable to access the stored information to determine whether said flexible operating system is being used as a native or as a virtualized operating system.

39. The system of claim 38 wherein if determined that it is being used as a virtualized operating system, said flexible operating system acting as a virtualized operating system.

40. The system of claim 38 wherein if determined that it is being used as a native operating system, said flexible operating system acting in a first manner, and if determined that it is being used as a virtualized operating system, said flexible operating system acting in a second manner.

41. The system of claim 40 wherein said first manner comprises acting as a native operating system, and wherein said second manner comprises acting as a paravirtualized operating system.